

HfS, HYPERFINE STRUCTURE FITTING TOOL

ROBERT ESTALELLA[†]Departament de Física Quàntica i Astrofísica (formerly Astronomia i Meteorologia), Institut de Ciències del Cosmos (ICC),
Universitat de Barcelona (IEEC-UB), Martí i Franquès 1, E08028 Barcelona, Spain(Dated: August 16, 2016)
accepted by PASP

ABSTRACT

HfS is a tool to fit the hyperfine structure of spectral lines, with multiple velocity components. The *HfS_nh3* procedures included in *HfS* fit simultaneously the hyperfine structure of the NH₃ (J, K) = (1, 1) and (2, 2) transitions, and perform a standard analysis to derive T_{ex} , NH₃ column density, T_{rot} , and T_{k} . *HfS* uses a Monte Carlo approach for fitting the line parameters. Especial attention is paid to the derivation of the parameter uncertainties. *HfS* includes procedures that make use of parallel computing for fitting spectra from a data cube.

Subject headings: ISM: molecules, methods: data analysis

1. INTRODUCTION

HfS (**H**yperfine **S**tructure) (Estalella 2016) is a tool to fit the hyperfine structure of spectral lines, with the possibility of fitting simultaneously multiple velocity components. The *HfS_nh3* procedures included in *HfS* fit simultaneously the hyperfine quadrupole and magnetic structure of the NH₃ (J, K) = (1, 1) and (2, 2) inversion transitions. The assumptions made by *HfS* are that the beam filling factor f , the excitation temperature T_{ex} , the hyperfine lines linewidth ΔV , and the central velocity V_{LSR} are the same for all the hyperfine lines.

For *HfS_nh3* these assumptions hold for the hyperfine lines of both NH₃ inversion transitions, (1, 1) and (2, 2). In addition, the results of the fit are used by *HfS_nh3* to derive physical parameters including the excitation temperature, NH₃ column density, rotational and kinetic temperature, with the assumption that the emitting region is homogeneous along the line of sight.

HfS is written in Fortran 77 and 90/95 (mainly for the dynamical storage of arrays). The graphic interface uses the PGplot Graphics Subroutine Library², and some procedures have a multiprocessor version running under Open MPI³. The interactive procedures are menu driven, allowing to select options with the keyboard, and for some actions, with the mouse cursor and buttons.

One of the advantages of *HfS* when compared with other packages performing hyperfine fitting, like for instance, the widely used CLASS of GILDAS⁴ is that it is well documented, easy to install, and with a simple interface. *HfS* can fit, in a single run, multiple velocity components to the spectra of a FITS data cube, and obtain the maps of the parameters fitted and derived from the fit, in a short computing time taking advantage of the multiple processors of current computers.

A preliminary version of *HfS* was briefly described and used for fitting NH₃(1, 1) and (2, 2) spectra in Sánchez-Monge et al. (2013).

The structure of the paper is as follows: The fit parameters used by *HfS* are described in §2, the fitting strategy in §3, the calculation of the synthetic spectrum in §4, the line param-

eters derived from the fit parameters in §5, the NH₃ physical parameters derived by *HfS_nh3* in §6, the error estimation for the fit parameters in §7 and for the derived parameters in §8. A comparison with the results obtained with CLASS and the Rosolowsky et al. (2008) routine is presented in §9. The different procedures that compose *HfS* are described in §10. Finally, in several Appendices we describe the requisites and installation instructions (Appendices A to C), and examples of use of *HfS*, and of input and output files (Appendices D to G).

2. FIT PARAMETERS

The general *HfS* procedures fit simultaneously, for every velocity component of a transition with hyperfine structure, four independent parameters,

- ΔV , hyperfine lines linewidth, assumed to be the same for all the hyperfine lines,
- V_{LSR} , main line central LSR velocity,
- $A_m^* \equiv A(1 - \exp\{-\tau_m\})$, main line peak intensity (for hyperfine lines wider than the hyperfine separation and the channel width), where A is the amplitude (see §5),
- $\tau_m^* \equiv 1 - \exp\{-\tau_m\}$, where τ_m is the optical depth of the main line.

The *HfS_nh3* procedures included in *HfS* fit simultaneously the hyperfine structure of a pair of spectra of the NH₃ inversion transitions (J, K) = (1, 1) and (2, 2) (see Table 1).

Differences in central velocity of the order of a tenth of km s⁻¹ are usually found between the (1, 1) and (2, 2) emissions (see for instance Sepúlveda et al. 2011). Thus, a different V_{LSR} is fitted for the (1, 1) and the (2, 2) spectra, resulting in two additional parameters for the (2, 2) transition, fitted simultaneously with the four parameters for the (1, 1) transition, ΔV , V_{LSR1} , A_{1m}^* , τ_{1m}^* , already described,

- V_{LSR2} , central LSR velocity of the (2, 2) transition,
- $A_{2m}^* \equiv A(1 - \exp\{-\tau_{2m}\})$, peak intensity of the (2, 2) main line (for hyperfine lines wider than the hyperfine separation and the channel width).

[†] The ICC (UB) is a CSIC-Associated Unit through the ICE (CSIC)

² <http://www.astro.caltech.edu/~tjp/pgplot/>

³ <http://www.open-mpi.org>

⁴ <https://www.iram.fr/IRAMFR/GILDAS/>

For each set of fit parameters, the optical depth of the (2, 2) main line is obtained from the relation

$$\tau_{2m}^* \equiv 1 - \exp\{-\tau_{2m}\} = \tau_{1m}^* \frac{A_{2m}^*}{A_{1m}^*}. \quad (1)$$

TABLE 1

RELATIVE VELOCITY V_{hyp} , AND OPTICAL DEPTH τ_{hyp} WITH RESPECT TO THAT OF THE MAIN LINE, τ_m , OF THE HYPERFINE LINES OF THE NH_3 (J, K) = (1, 1) AND (2, 2) INVERSION TRANSITIONS (MANGUM & SHIRLEY 2015). (OS: OUTER SATELLITE, IS: INNER SATELLITE, M: MAIN)

	$(J, K) = (1, 1)$		$(J, K) = (2, 2)$	
	V_{hyp} (km s ⁻¹)	τ_{hyp}	V_{hyp} (km s ⁻¹)	τ_{hyp}
os	-19.548593	0.148148	-26.557749	0.004186
	-19.409429	0.074074	-26.042011	0.037674
			-25.981267	0.020930
is	-7.815393	0.166667	-16.401673	0.037209
	-7.373698	0.018519	-16.389235	0.026047
	-7.234546	0.092593	-15.873510	0.001860
m	-0.252188	0.033333	-0.589779	0.020930
	-0.213155	0.092593	-0.531971	0.010631
	-0.133040	0.466667	-0.502427	0.011628
	-0.073991	0.018519	-0.013336	0.146512
	0.189495	0.300000	-0.003910	0.499668
	0.308643	0.033333	0.013298	0.267442
	0.323117	0.018519	0.524366	0.011628
	0.462268	0.037037	0.529035	0.010631
is	7.350050	0.018519	0.563171	0.020930
	7.469198	0.166667	15.883012	0.001860
	7.886323	0.092593	16.398737	0.026047
os	19.319597	0.148148	16.411175	0.037209
	19.845140	0.074074	25.981267	0.020930
			26.042011	0.037674
			26.557749	0.004186
Sum		2.000000		1.255814

The use of $A_m^* \equiv A(1 - \exp\{-\tau_m\})$ and $\tau_m^* \equiv 1 - \exp\{-\tau_m\}$ as fit parameters instead of A or $A\tau_m$, and τ_m needs some justification. First, A_m^* is well determined even in the two limiting cases, $\tau_m \ll 1$ (with A ill-determined) and $\tau_m \gg 1$ (with $A\tau_m$ ill-determined). This is because A_m^* is a true physical magnitude, i.e. the peak intensity of the main line, provided that the hyperfine lines that may compose the main line are blended. Second, τ_m^* is bounded, $0 < \tau_m^* < 1$, and the two bounds correspond to the optically thin and thick limiting cases, where the fit is degenerate, and no longer depends on τ_m . In short, the two fit parameters are well determined in all cases, even in the optically thin and thick limits. The only drawback of the use of τ_m^* as fit parameter is for the case of extremely high opacity in the simultaneous fit of $\text{NH}_3(1, 1)$ and (2, 2), when the intensity of the (2, 2) main and satellite lines is similar. In its present version, *HfS_nh3* is not able to deal accurately with $\tau_{1m} > 16$, but this case is indeed very rare. In a survey of low-mass cores in Perseus (Rosolowsky et al. 2008), none of the sources has such a high value of optical depth, and in another survey of high-mass clumps in the Galactic Plane (Svoboda et al. 2016) only 1% of the sources show $\tau_{1m} > 16$. Thus, *HfS_nh3* should work for the vast majority of cases.

The fitting procedure ends with a set of four or, for the *HfS_nh3* procedures, six values plus τ_{2m}^* , of the fit param-

eters for every velocity component, which minimize the fit residual χ^2 of the spectrum (see §3), and an estimation of the uncertainty of the fit parameters (see §7), $\sigma(\Delta V)$, $\sigma(V_{\text{LSR}})$, $\sigma(A_m^*)$, $\sigma(\tau_m^*)$, and, for the *HfS_nh3* procedures, $\sigma(V_{\text{LSR}2})$, and $\sigma(A_{2m}^*)$.

3. FITTING STRATEGY

The fitting procedure is similar to that used in other fitting problems by Estalella et al. (2012) and Palau et al. (2014). *HfS* samples the space parameter of dimension m (four or six times the number of velocity components), defined by the parameters p_1, \dots, p_m , to find the minimum value of the fit residual χ^2 ,

$$\chi^2 = \sum_{i=1}^N \left[\frac{y_i^{\text{obs}} - y_i^{\text{mod}}(p_1, \dots, p_m)}{\sigma_i} \right]^2, \quad (2)$$

where y_i^{obs} are the observed line intensities, for a total of N spectral channels, $y_i^{\text{mod}}(p_1, \dots, p_m)$ are the model line intensities, depending on m free parameters, and σ_i are the errors of the observations.

The sampling strategy is based on that used in AGA (Asexual Genetic Algorithm) (Cantó, Curiel, Martínez-Gómez 2009). The procedure starts with a number of samples of the parameter space within the initial search range for each parameter (seeds). For each seed, the residual χ^2 is computed for a number of samples (descendants) within the search range centered on the seed value. The best samples, i.e. those with the lowest χ^2 , are kept as seeds for the next loop, for which the search ranges are decreased by a constant factor. The procedure is iterated, and is stopped after a given number of loops.

Several sampling methods of the m -dimensional parameter space are possible, i.e. regular grid, random, Halton or Sobol pseudo-random sequences (Halton 1964; Sobol 1967). *HfS* uses a Sobol pseudo-random sequence because it samples the m -dimensional parameter space more evenly than a purely random sequence, even for high values of m , and the convergence of the fitting procedure to the minimum of χ^2 is faster.

4. SYNTHETIC SPECTRUM

Let us assume that the transition being fitted has n_h hyperfine lines, and for each hyperfine $j = 1, \dots, n_h$, V_{hyp}^j is the velocity shift with respect to the main line, and τ_{hyp}^j is the ratio of the optical depth of the hyperfine and that of the main line, τ_m . Let n_c be the number of velocity components fitted, and for each velocity component $i = 1, \dots, n_c$, ΔV^i , V_{LSR}^i , A^i , and τ_m^i are the values of the fit or derived line parameters (see §5).

For each velocity channel $k = 1, \dots, N$, with central velocity V_k and channel width ΔV_{ch} , the contribution of the velocity component i to the optical depth is calculated as

$$\tau_k^i = \sum_{j=1}^{n_h} \frac{\tau_m^i \tau_{\text{hyp}}^j}{\Delta V_{\text{ch}}} G, \quad (3)$$

where G is the integral over the velocity range of the channel,

$$G = \int_{V_k - \Delta V_{\text{ch}}/2}^{V_k + \Delta V_{\text{ch}}/2} e^{-4 \ln 2 [(v - V_{\text{LSR}}^i - V_{\text{hyp}}^j)/\Delta V^i]^2} dv. \quad (4)$$

G is evaluated by means of the error function $\text{erf}(x) =$

$2/\sqrt{\pi} \int_0^x \exp(-t^2) dt$, as

$$G = \frac{\sqrt{\pi}}{4\sqrt{\ln 2}} \Delta V^i [\operatorname{erf}(x^+) - \operatorname{erf}(x^-)], \quad (5)$$

where

$$x^\pm = 2\sqrt{\ln 2} \frac{V_k \pm \Delta V_{\text{ch}}/2 - V_{\text{LSR}}^i - V_{\text{hyp}}^j}{\Delta V^i}. \quad (6)$$

If the difference between x^+ and x^- is too low, within less than 1 part in 10^4 , Equation 5 can produce a large roundoff error, and G is evaluated as

$$G = \frac{\Delta V^i}{2\sqrt{\ln 2}} (x^+ - x^-) e^{-[(x^+ + x^-)/2]^2}. \quad (7)$$

Finally, the intensity of the channel k of the synthetic spectrum is calculated as the sum of intensities of the different velocity components,

$$T_k = \sum_{i=1}^{n_c} A^i (1 - e^{-\tau_k^i}). \quad (8)$$

5. DERIVED LINE PARAMETERS

From the values of the fit parameters, the following derived line parameters are calculated, which are necessary for calculating the synthetic spectra and the estimation of parameters with physical interest:

A, amplitude. — The amplitude is

$$A = f[J_\nu(T_{\text{ex}}) - J_\nu(T_{\text{bg}})], \quad (9)$$

where f is the beam filling factor, T_{ex} the excitation temperature, T_{bg} the background temperature, and

$$J_\nu(T) = \frac{h\nu/k}{e^{h\nu/kT} - 1}, \quad (10)$$

is the Planck-corrected temperature. A is calculated from

$$A = \frac{A_m^*}{\tau_m^*}. \quad (11)$$

In the case of *HfS_nh3*, the excitation temperature T_{ex} and the filling factor f are assumed to be the same for the (1, 1) and (2, 2) lines. Since the frequencies of both transitions are very close (see Table 2), these assumptions imply that the amplitude A is the same for both lines, within less than 1 in 10^4 .

τ_m , optical depth of the main line. — Calculated from

$$\tau_m = -\ln(1 - \tau_m^*). \quad (12)$$

Special care has to be taken when $\tau_m \simeq \tau_m^* \ll 1$, since the last expression involves the difference of 1 and a number near 1. In this case, a good approximation is the Taylor expansion

$$\tau_m \simeq \tau_m^* + \frac{1}{2}\tau_m^{*2} + \frac{1}{3}\tau_m^{*3}. \quad (13)$$

In the case of *HfS_nh3*, τ_{1m} and τ_{2m} are derived in the same way.

$A\tau_m$, amplitude times the main line optical depth. — Calculated from

$$A\tau_m = A_m^* \frac{\tau_m}{\tau_m^*}. \quad (14)$$

Note that for $\tau_m^* \ll 1$, $A\tau_m \simeq A_m^*$. In the case of *HfS_nh3*, $A\tau_{1m}$ and $A\tau_{2m}$ are derived in the same way.

6. NH₃ DERIVED PHYSICAL PARAMETERS

In addition to the derived line parameters, *HfS_nh3* derives physical parameters using the standard analysis of NH₃ (1, 1) and (2, 2) observations, which assumes that the region observed is homogeneous along the line of sight. The physical parameters derived are the excitation temperature T_{ex} , the NH₃ (1, 1) and (2, 2) beam-averaged column densities $fN(1, 1)$ and $fN(2, 2)$, the rotational temperature T_{rot} , the NH₃ beam-averaged column density $fN(\text{NH}_3)$, and the kinetic temperature T_k .

Excitation temperature. — The excitation temperature is obtained from the amplitude A ,

$$T_{\text{ex}} = \frac{T_\nu}{\ln \left[1 + \frac{T_\nu}{A/f + J_\nu(T_{\text{bg}})} \right]}, \quad (15)$$

where $T_\nu = h\nu/k$ is the frequency in temperature units. The frequency of the (1, 1) transition is used, but since the frequency of the (2, 2) transition is very close (see Table 2), the result does not depend on which of the two frequencies is used. For values of $T_{\text{ex}} \gg T_{\text{bg}} > T_\nu$, this expression simplifies to

$$T_{\text{ex}} \simeq A/f. \quad (16)$$

The value of the excitation temperature depends on the value assumed for the filling factor f . The usual assumption is that the filling factor $f = 1$. The value obtained with this assumption is a lower limit for the value of T_{ex} . On the contrary, if we assume that $f \ll 1$, $T_{\text{ex}} \rightarrow \infty$.

NH₃ (1, 1) and (2, 2) column densities. — The column density of the $(J, K) = (1, 1)$ and $(2, 2)$ levels (i.e. the sum of column densities of the two inversion levels of the corresponding (J, K) rotational level) can be given as (Anglada et al. 1995; Estalella & Anglada 1997)

$$N(J, K) = \sqrt{\frac{\pi}{4 \ln 2}} \frac{8\pi\nu_{jk}^3}{c^3 A_{jk}} R_m \frac{\exp(T_\nu/T_{\text{ex}}) + 1}{\exp(T_\nu/T_{\text{ex}}) - 1} \tau_m \Delta V, \quad (17)$$

where A_{jk} is the Einstein coefficient of the inversion transition of the rotational level (J, K) , $R_m = \tau_{\text{tot}}/\tau_m$ is the ratio of total and main line optical depths of the inversion transition, and $T_\nu = h\nu_{JK}/k$ is the frequency of the inversion transition in temperature units (see Table 2).

The last expression depends on the value of the filling factor f assumed to derive T_{ex} . The explicit dependence on f is

$$\frac{\exp(T_\nu/T_{\text{ex}}) + 1}{\exp(T_\nu/T_{\text{ex}}) - 1} = \frac{2}{fT_\nu} (A + f[J_\nu(T_{\text{bg}}) + T_\nu/2]), \quad (18)$$

so that the beam-averaged column density can be expressed as

$$fN(J, K) = \sqrt{\frac{\pi}{4 \ln 2}} \frac{16\pi k\nu_{jk}^2}{hc^3 A_{jk}} R_m \times \left(A\tau_m + f \left[J_\nu(T_{\text{bg}}) + \frac{T_\nu}{2} \right] \tau_m \right) \Delta V. \quad (19)$$

The maximum value of $fN(J, K)$ is obtained for $f = 1$ (the usual assumption to derive T_{ex}), while the minimum value is obtained for $f \ll 1$. In the latter case (or for $A \simeq T_{\text{ex}} \gg T_{\text{bg}} > T_\nu$), the expression simplifies to

$$fN(J, K) = \sqrt{\frac{\pi}{4 \ln 2}} \frac{16\pi k\nu_{jk}^2}{hc^3 A_{jk}} R_m A\tau_m \Delta V. \quad (20)$$

TABLE 2

VALUES OF THE NH_3 (1, 1) AND (2, 2) INVERSION TRANSITION FREQUENCIES (KUKOLICH 1967), SPONTANEOUS EMISSION EINSTEIN COEFFICIENTS (OSORIO ET AL. 2009), RATIOS OF TOTAL TO MAIN LINE OPTICAL DEPTHS (MANGUM & SHIRLEY 2015), AND $B(J, K)$ AND $C(J, K)$ COEFFICIENTS OF ANGLADA ET AL. (1995) RECALCULATED WITH THE IMPROVED VALUES OF THE CONSTANTS IN THIS TABLE.

(J, K)	ν_{jk} (GHz)	$T_v = h\nu_{jk}/k$ (K)	A_{jk} (10^{-7} s^{-1})	$R_m =$ τ_{tot}/τ_m	$B(J, K)$	$C(J, K)$
(1, 1)	23.69450	1.137157	1.66838	2.000000	1.58339×10^{13}	2.78482×10^{13}
(2, 2)	23.72263	1.138507	2.23246	1.255814	7.45665×10^{12}	1.30990×10^{13}

The values of the constants appearing in these equations are given in Table 2.

In practical units the two equations become (Anglada et al. 1995)

$$\left[\frac{N(J, K)}{\text{cm}^2} \right] = B(J, K) \frac{\exp(T_v/T_{\text{ex}}) + 1}{\exp(T_v/T_{\text{ex}}) - 1} \tau_m \left[\frac{\Delta V}{\text{km s}^{-1}} \right] \quad (21)$$

and, for $f \ll 1$, or $T_{\text{ex}} \gg T_{\text{bg}} > T_v$,

$$\left[\frac{fN(J, K)}{\text{cm}^2} \right] = C(J, K) A \tau_m \left[\frac{\Delta V}{\text{km s}^{-1}} \right], \quad (22)$$

The values of the constants $B(J, K)$ and $C(J, K)$ for the (1, 1) and (2, 2) transitions are given in Table 2. The expression equivalent to these two equations, with an explicit dependence on f , is,

$$\left[\frac{fN(J, K)}{\text{cm}^2} \right] = C(J, K) \left(A \tau_m + f \left[J_v(T_{\text{bg}}) + \frac{T_v}{2} \right] \tau_m \right) \left[\frac{\Delta V}{\text{km s}^{-1}} \right]. \quad (23)$$

Rotational temperature. — The rotational temperature is obtained from the ratio of (1, 1) and (2, 2) column densities,

$$T_{\text{rot}} = \frac{(E_{22} - E_{11})/k}{\ln \left(\frac{g_{22} N(1, 1)}{g_{11} N(2, 2)} \right)}, \quad (24)$$

where E_{11} , E_{22} , and g_{11} , g_{22} , are the energies and degeneracies of the corresponding levels. In practical units the equation becomes (see Table 3),

$$\left[\frac{T_{\text{rot}}}{\text{K}} \right] = \frac{40.99}{\ln \left(\frac{5}{3} \frac{N(1, 1)}{N(2, 2)} \right)}. \quad (25)$$

TABLE 3

DEGENERACIES AND ENERGIES ABOVE THE (1, 1) LEVEL OF THE LOWER METASTABLES LEVELS OF NH_3 (POYNTER & KAKAR 1975; MANGUM & SHIRLEY 2015). NOTE THAT THE VALUES OF THE ENERGIES REPORTED BY THESE AUTHORS ARE SLIGHTLY DIFFERENT FROM THOSE GIVEN IN HO & TOWNES (1983).

(J, K)	g_{JK}	$(E_{JK} - E_{11})/k$ (K)
(0, 0)	1/3	-22.64
(1, 1)	1	0.00
(2, 2)	5/3	40.99
(3, 3)	14/3	99.76

NH_3 column density. — The ammonia total column density is usually estimated assuming that the population ratios between pairs of levels are given by the same rotational temperature (CTEX approximation), and that only the metastable levels

$J = K$, up to $(J, K) = (3, 3)$, are populated. While this assumption is reasonable for low-mass dense cores with moderate temperatures, it is not appropriate for hot cores, where NH_3 inversion transitions (6, 6) to (14, 14) are detected (see for instance Goddi, Zhang, & Moscadelli 2015).

In addition, since the $\text{NH}_3(1, 1)$ and (2, 2) levels correspond to para- NH_3 , no information on the ortho- NH_3 is obtained from the (1, 1) and (2, 2) spectra. Thus, an ortho-to-para ratio has to be assumed. The usual assumption is to take an ortho-to-para ratio of 1, although a value of ~ 0.7 has been observed in some star-forming regions (see for instance the discussion in Faure et al. 2013). We will assume an ortho-to-para ratio of 1.

With the former assumptions,

$$N(\text{NH}_3) = N(1, 1) Q, \quad (26)$$

with the partition function Q given by

$$Q = \sum_{J,K=0}^3 \frac{g_{JK}}{g_{11}} e^{(E_{11}-E_{JK})/kT_{\text{rot}}}. \quad (27)$$

In practical units (see Table 3),

$$N(\text{NH}_3) = N(1, 1) \left[\frac{1}{3} e^{22.64/T_{\text{rot}}} + 1 + \frac{5}{3} e^{-40.99/T_{\text{rot}}} + \frac{14}{3} e^{-99.76/T_{\text{rot}}} \right]. \quad (28)$$

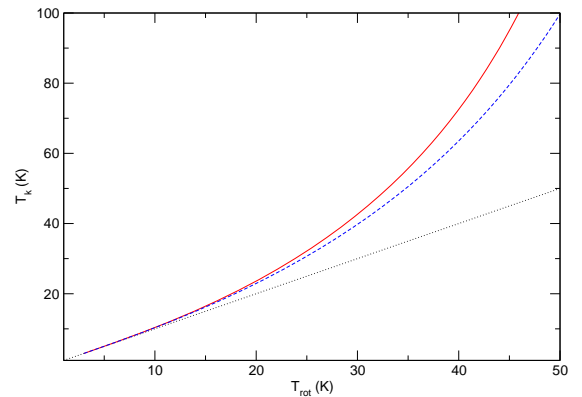


FIG. 1.— T_k as a function of T_{rot} . Red continuum line: Maret et al. (2009) and present work; blue dashed line: Swift et al. (2005), Rosolowsky et al. (2008), Mangum & Shirley (2015); black dotted line: $T_k = T_{\text{rot}}$.

Kinetic temperature. — The kinetic temperature can be taken to be equal to the rotational temperature T_{rot} , but a better estimation can be given taking into account collisional transitions to other levels. A 3-level approximation considering only the rotational levels (1, 1), (2, 2), and (2, 1) can be shown to be (Walmsley & Ungerechts 1983; Danby et al. 1988; Mangum

& Shirley 2015)

$$T_{\text{rot}} = \frac{T_k}{1 + \frac{T_k}{T_0} \ln \left(1 + \frac{C(22 \rightarrow 21)}{C(22 \rightarrow 11)} \right)}, \quad (29)$$

where $T_0 = (E_{22} - E_{11})/k$, and $C(22 \rightarrow 21)$ and $C(22 \rightarrow 11)$ are the collisional excitation and desexcitation rates between the corresponding levels. Calculations involving more rotational levels were performed by Danby et al. (1988), and more recently by Maret et al. (2009) using improved values of the collisional rates. A good approximation to the results of Maret et al. (2009) can be given as a slightly modified version of the widely used expression (Swift et al. 2005; Rosolowsky et al. 2008; Mangum & Shirley 2015, see Fig. 1), which is

$$T_{\text{rot}} = \frac{T_k}{1 + \frac{T_k}{T_0} \ln [1 + 0.73 e^{-(E_{21} - E_{22})/kT_k}]}, \quad (30)$$

with $T_0 = 40.99$ K, $(E_{21} - E_{22})/k = 16.26$ K, and the numerical value 0.73 was determined to fit the results shown in Fig. 5 of Maret et al. (2009). Note that this relation implies that T_{rot} is always below a value $T_{\text{rot}} = 40.99 / \ln 1.73 = 74.8$ K (see Fig. 1). Given a value of T_{rot} , the implicit equation must be solved to find T_k . A possible iterative algorithm to solve the equation is

$$T_k^{(n+1)} = T_{\text{rot}} \left[1 + \frac{T_k^{(n)}}{40.99} \ln \left(1 + 0.73 e^{-16.26/T_k^{(n)}} \right) \right], \quad (31)$$

starting with $T_k^{(0)} = T_{\text{rot}}$.

7. ERROR ESTIMATION OF THE FIT PARAMETERS

7.1. Parameter-space confidence region

Let us assume that the best fit to the observed data is obtained for values p_k^0 of the parameters, for which the residual χ^2 is minimum,

$$\chi_{\min}^2 = \sum_{i=1}^N \left[\frac{y_i^{\text{obs}} - y_i^{\text{mod}}(p_1^0, \dots, p_m^0)}{\sigma_i} \right]^2, \quad (32)$$

where σ_i is the error of y_i^{obs} . The uncertainty, $\sigma(p_k)$, in the values derived for the parameters p_k can be estimated as the projection of the confidence region of the m -dimensional space parameter for which χ^2 does not exceed the minimum value by an amount $\Delta(m, \alpha)$, where α is the significance level ($0 < \alpha < 1$). Following Avni (1976) and Wall & Jenkins (2003), the probability

$$\text{Prob} [\chi^2 - \chi_{\min}^2 \leq \Delta(m, \alpha)] = \alpha, \quad (33)$$

is that of a chi-square distribution with m degrees of freedom. Thus, $\Delta(m, \alpha)$ is the increment of χ^2 such that if the observation is repeated a large number of times, a fraction α of times the values of the parameters fitted will be inside the confidence region, i.e. in the interval $p_k \pm \sigma(p_k)$. The values of $\Delta(m, \alpha)$ for significance levels equivalents to 1, 2, and 3 sigmas for a Gaussian error distribution, and different values of m are shown in Table 4.

Assuming that the model fits well the observations, i.e. $\chi_{\min}^2 \simeq N - m$, the condition

$$\chi^2 = \chi_{\min}^2 + \Delta(m, \alpha), \quad (34)$$

TABLE 4
VALUES OF $\Delta(m, \alpha)$ FOR CALCULATING THE PARAMETER UNCERTAINTIES, WHERE m IS THE NUMBER OF PARAMETERS FITTED SIMULTANEOUSLY, AND α IS THE SIGNIFICANCE LEVEL, GIVEN IN PERCENT AND IN THE EQUIVALENT NUMBER OF SIGMAS FOR A GAUSSIAN ERROR DISTRIBUTION. THE VALUES SHOWN ARE FOR VALUES OF m MULTIPLE OF 4 OR 6, USED BY HfS.

m	α		
	68.27% (1 σ)	95.45% (2 σ)	99.73% (3 σ)
4	4.72	9.72	16.25
6	7.04	12.85	20.06
8	9.30	15.79	23.57
12	13.74	21.35	30.09
16	18.11	26.65	36.21
18	20.28	29.24	39.17
20	22.44	31.80	42.08
24	26.73	36.83	47.76
28	31.00	41.78	53.31
30	33.12	44.22	56.04
32	35.24	46.66	58.75
36	39.48	51.48	64.10
42	45.80	58.64	71.99
48	52.11	65.72	79.75
54	58.39	72.72	87.41

can be written as

$$\frac{\chi^2}{\chi_{\min}^2} \simeq 1 + \frac{\Delta(m, \alpha)}{N - m}. \quad (35)$$

This expression is useful since it can be given in terms of the weighted rms fit residual, σ ,

$$\sigma^2 = \frac{\sum_{i=1}^N [(y_i^{\text{obs}} - y_i^{\text{mod}})/\sigma_i]^2}{\sum_{i=1}^N 1/\sigma_i^2} = \frac{\chi^2}{\sum_{i=1}^N 1/\sigma_i^2}, \quad (36)$$

for which we obtain that the confidence region is given by the parameter values that increase the rms fit residual to

$$\sigma \simeq \sigma_{\min} \sqrt{1 + \frac{\Delta(m, \alpha)}{N - m}}. \quad (37)$$

This last expression can be used even when the errors of the observations are unknown.

7.2. Modeling the fit residual χ^2

7.2.1. Quadratic approximation of the fit residual

Let us assume that around its minimum value, the residual χ^2 can be approximated by a quadratic function,

$$\chi^2 \simeq \chi_{\min}^2 + \sum_{i,j=1}^m a_{ij} x_i x_j + 2 \sum_{i=1}^m b_i x_i, \quad (38)$$

where $x_j = p_j - p_j^0$ are the increment of the parameter values from their best-fit values. The confidence region of the m -dimensional parameter space will be the region inside the surface

$$\sum_{i,j=1}^m a_{ij} x_i x_j + 2 \sum_{i=1}^m b_i x_i = \Delta(m, \alpha), \quad (39)$$

which is the general equation of a m -dimensional quadric (McConnell 2011). This equation depends on a total of $m(m+3)/2$ coefficients: $m(m+1)/2$ symmetric a_{ij} coefficients ($a_{ij} = a_{ji}$), and m coefficients b_i .

In array form, the quadric equation can be expressed as

$$\begin{pmatrix} x_1 & \dots & x_m & 1 \end{pmatrix} \begin{pmatrix} a_{11} & \dots & a_{1m} & b_1 \\ \vdots & \ddots & \vdots & \vdots \\ a_{m1} & \dots & a_{mm} & b_m \\ b_1 & \dots & b_m & c \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_m \\ 1 \end{pmatrix} = 0, \quad (40)$$

with $c = -\Delta(m, \alpha)$, or, with the obvious definitions for arrays \mathbf{A} , \mathbf{B} , and \mathbf{X} ,

$$\begin{pmatrix} \mathbf{X}^t & 1 \end{pmatrix} \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^t & c \end{pmatrix} \begin{pmatrix} \mathbf{X} \\ 1 \end{pmatrix} = 0. \quad (41)$$

In order to estimate the uncertainties of the fit parameters, $\sigma(p_k)$, we have to calculate the projections of the quadric onto each axis k . If the fit residual is well behaved, we may expect that the quadric is an ellipsoid, i.e. its projections onto the plane defined by any pair of parameters is an ellipse, and the ellipsoid has finite projections onto any axis.

7.2.2. Projections of an ellipsoid

The projections of the ellipsoid onto each coordinate axis can be found as the intersections of the hyperplane perpendicular to the axis, tangent to the ellipsoid. The equation of the tangent hyperplane at a point (x_1^0, \dots, x_m^0) of the ellipsoid is given by (McConnell 2011)

$$\begin{pmatrix} x_1^0 & \dots & x_m^0 & 1 \end{pmatrix} \begin{pmatrix} a_{11} & \dots & a_{1m} & b_1 \\ \vdots & \ddots & \vdots & \vdots \\ a_{m1} & \dots & a_{mm} & b_m \\ b_1 & \dots & b_m & c \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_m \\ 1 \end{pmatrix} = 0, \quad (42)$$

or

$$\sum_{i,j=1}^m a_{ij} x_i^0 x_j + \sum_{j=1}^m b_j x_j + \sum_{i=1}^m b_i x_i^0 + c = 0. \quad (43)$$

The equation of an hyperplane perpendicular to the k axis is $x_k = \text{constant}$, so that the equation of the tangent hyperplane is

$$\left[\sum_{i=1}^m a_{ik} x_i^0 + b_k \right] x_k + \sum_{i=1}^m b_i x_i^0 + c = 0, \quad (44)$$

and the coefficients of x_j for $j \neq k$ have to be zero,

$$\sum_{i=1}^m a_{ij} x_i^0 + b_j = 0 \quad (j \neq k). \quad (45)$$

Eqs. 45 form a system of $m - 1$ linear equations with m unknowns, the coordinates of the tangent point. We can consider that x_j^0 ($j \neq k$) are the $m - 1$ unknowns of the system, which can be derived as a function of x_k^0 . The system of equations can be written with arrays of size m ,

$$\mathbf{A}_k \mathbf{X} = \mathbf{Z}_k x_k^0 + \mathbf{C}_k \quad (46)$$

where \mathbf{A}_k is the array \mathbf{A} with zeros in the k row and column, $a_{ik} = a_{ki} = 0$, and 1 in the diagonal term, $a_{kk} = 1$, and \mathbf{Z}_k and

\mathbf{B}_k have 1 and 0 respectively in the k row,

$$\mathbf{A}_k = \begin{pmatrix} a_{11} & \dots & 0 & \dots & a_{1m} \\ \vdots & & \vdots & & \vdots \\ 0 & \dots & 1 & \dots & 0 \\ \vdots & & \vdots & & \vdots \\ a_{m1} & \dots & 0 & \dots & a_{mm} \end{pmatrix}, \quad \mathbf{Z}_k = \begin{pmatrix} -a_{1k} \\ \vdots \\ 1 \\ \vdots \\ -a_{mk} \end{pmatrix}, \quad \mathbf{C}_k = \begin{pmatrix} -b_1 \\ \vdots \\ 0 \\ \vdots \\ -b_m \end{pmatrix}. \quad (47)$$

From the system we can derive the solution giving x_j^0 in terms of x_k^0 ,

$$\mathbf{X} = \mathbf{D}_k x_k^0 + \mathbf{E}_k, \quad \text{with} \quad \mathbf{D}_k = \mathbf{A}_k^{-1} \mathbf{Z}_k, \quad \mathbf{E}_k = \mathbf{A}_k^{-1} \mathbf{C}_k, \quad (48)$$

which can be expressed as

$$x_j^0 = d_{jk} x_k^0 + e_{jk} \quad (j = 1, \dots, m), \quad (49)$$

where $d_{kk} = 1$ and $e_{kk} = 0$.

The tangent point must fulfill the tangent hyperplane equation, Eq. 44. By substitution of Eq. 49 in the tangent hyperplane equation, and setting $x_k = x_k^0$, we get a second degree equation in x_k^0 ,

$$\left[\sum_{i=1}^m a_{ki} d_{ik} \right] (x_k^0)^2 + \left[\sum_{i=1}^m (a_{ki} e_{ik} + b_i d_{ik}) + b_k \right] x_k^0 + \left[\sum_{i=1}^m b_i e_{ik} + c \right] = 0. \quad (50)$$

The two solutions of the equation provide the two k -coordinates of the projections of the ellipsoid onto the k axis.

7.2.3. Case of a centered ellipsoid

Since we are assuming that we know that the minimum residual χ^2 is well determined, the quadratic function of Eq. 51 must have a minimum at the origin, and the linear terms in the variables vanish because the partial derivatives at the origin must be zero. This means that we can assume that the quadric is centered, and its equation becomes

$$\sum_{i,j=1}^m a_{ij} x_i x_j + c = 0. \quad (51)$$

In this case, the equation of the tangent plane is simpler, since $\mathbf{B} = 0$, $\mathbf{C}_k = 0$, and $\mathbf{E}_k = 0$. The second degree equation (Eq. 50) becomes

$$\left[\sum_{i=1}^m a_{ki} d_{ik} \right] (x_k^0)^2 + c = 0, \quad (52)$$

and the projections are symmetric, $\pm x_k^0$.

7.2.4. Coefficients of the ellipsoid

The centered ellipsoid depends on $m(m+1)/2$ symmetric a_{ij} coefficients. Let us examine a sufficient number of constraints to derive these coefficients.

Diagonal coefficients a_{ii} . — For each parameter i we can obtain constraints from the values of the residual χ^2 increment for different increments x_i of the parameter p_i , while keeping the rest of parameters constant,

$$\chi^2(p_1^0, \dots, p_i^0 + x_i, \dots, p_m^0) = \chi_{\min}^2 + \Delta_i. \quad (53)$$

For each value of the increments of the parameter and the residual χ^2 increment, x_i^n , Δ_i^n , we have a constraint on a_{ii} ,

$$a_{ii}(x_i^n)^2 \simeq \Delta_i^n. \quad (54)$$

Although a single value is enough to determine a_{ii} , at least two are recommended, above and below the best-fit value, i.e. with $x_i > 0$ and $x_i < 0$. In general, the best approximation for a_{ii} (so that the sum of the squares of $[\Delta_i^n - a_{ii}(x_i^n)^2]$ is minimum) is given by

$$a_{ii} = \frac{\sum_n \Delta_i^n (x_i^n)^2}{\sum_n (x_i^n)^4} \quad (55)$$

where the sums are for all increments evaluated. Note that, provided that the Δ_i^n are positive, the diagonal term a_{ii} will always be positive. For a good characterization of the behavior of χ^2 , at least two values of Δ_i^n should be close to $\Delta(m, \alpha)$.

The intersections of the ellipsoid with the coordinate axis i are $\pm(\Delta(m, \alpha)/a_{ii})^{1/2}$. For the case of statistically independent parameters, these intersections will coincide with the projections of the ellipsoid, since the coordinate axes are the principal axes of the ellipsoid and the cross-terms of the ellipsoid vanish. However, in general, there will be some dependence among the parameters, and the cross terms will not be zero.

Cross-coefficients a_{ij} ($i \neq j$). — The constraints to derive the cross terms a_{ij} can be obtained from the value of the residual χ^2 for the simultaneous increment of the two parameters i and j , while keeping the rest of parameters constant,

$$\chi^2(p_1^0, \dots, p_i^0 + x_i, \dots, p_j^0 + x_j, \dots, p_m^0) = \chi_{\min}^2 + \Delta_{ij} \quad (56)$$

For each pair of increments of the parameters and the residual χ^2 increment, x_i^n , x_j^n , Δ_{ij}^n , we have a constraint on a_{ij} ,

$$a_{ii}(x_i^n)^2 + 2a_{ij}x_i^n x_j^n + a_{jj}(x_j^n)^2 \simeq \Delta_{ij}^n. \quad (57)$$

Taking into account Eq. 54, and defining $\delta_{ij}^n \equiv \Delta_{ij}^n - \Delta_i^n - \Delta_j^n$, we have

$$2a_{ij}x_i^n x_j^n \simeq \delta_{ij}^n \quad (58)$$

Although a single pair of values x_i , x_j , is enough to determine a_{ij} , at least four are recommended, above and below the best-fit value for each parameter, i.e. with the four combinations of $x_i > 0$, $x_i < 0$, $x_j > 0$, and $x_j < 0$. In general, the best approximation for a_{ij} (in the same sense as for the diagonal term) is given by

$$a_{ij} = \frac{1}{2} \frac{\sum_n \delta_{ij}^n x_i^n x_j^n}{\sum_n (x_i^n x_j^n)^2} \quad (59)$$

where the sums are for all the pairs of increments evaluated.

7.2.5. Practical case

A practical implementation of the procedure is as follows. Let us call

$$f(x_1, \dots, x_m) = \chi^2(p_1^0 + x_1, \dots, p_m^0 + x_m) - \chi_{\min}^2. \quad (60)$$

1. For each parameter p_i we estimate a positive and a negative increment, x_i^+ and x_i^- , such that the increment of the residual χ^2 is close to $\Delta(m, \alpha)$,

$$\begin{aligned} f(0, \dots, x_i^+, \dots, 0) &= \Delta_i^+ \simeq \Delta(m, \alpha), \\ f(0, \dots, x_i^-, \dots, 0) &= \Delta_i^- \simeq \Delta(m, \alpha). \end{aligned} \quad (61)$$

The diagonal coefficient a_{ii} (Eq. 54) is given by

$$a_{ii} = \frac{\Delta_i^+(x_i^+)^2 + \Delta_i^-(x_i^-)^2}{(x_i^+)^4 + (x_i^-)^4}. \quad (62)$$

2. For each pair of parameters p_i , p_j ($i \neq j$) we calculate

$$\begin{aligned} \delta_{ij}^{++} &= f(0, \dots, x_i^+, \dots, x_j^+, \dots, 0) - \Delta_i^+ - \Delta_j^+, \\ \delta_{ij}^{+-} &= f(0, \dots, x_i^+, \dots, x_j^-, \dots, 0) - \Delta_i^+ - \Delta_j^-, \\ \delta_{ij}^{-+} &= f(0, \dots, x_i^-, \dots, x_j^+, \dots, 0) - \Delta_i^- - \Delta_j^+, \\ \delta_{ij}^{--} &= f(0, \dots, x_i^-, \dots, x_j^-, \dots, 0) - \Delta_i^- - \Delta_j^-. \end{aligned} \quad (63)$$

The cross-coefficient a_{ij} (Eq. 59) is given by

$$a_{ij} = \frac{1}{2} \frac{\delta_{ij}^{++} x_i^+ x_j^+ + \delta_{ij}^{+-} x_i^+ x_j^- + \delta_{ij}^{-+} x_i^- x_j^+ + \delta_{ij}^{--} x_i^- x_j^-}{(x_i^+ x_j^+)^2 + (x_i^+ x_j^-)^2 + (x_i^- x_j^+)^2 + (x_i^- x_j^-)^2}. \quad (64)$$

3. For each parameter k we construct the arrays \mathbf{A}_k , \mathbf{Z}_k (Eq. 47), and solve the system of linear equations $\mathbf{A}_k \mathbf{D}_k = \mathbf{Z}_k$. Finally, we calculate the quadratic coefficient, $\sum_{i=1}^m a_{ki} d_{ik}$ (Eq. 52), and the projection $x_k^0 = [\Delta(m, \alpha) / \sum_{i=1}^m a_{ki} d_{ik}]^{1/2}$.

In some cases, the quadratic approximation of χ^2 is not good enough, and the determination of the projections can fail: for some parameter p_k the array \mathbf{A}_k may have null determinant, or the quadratic coefficient of Eq. 52 may be negative. In these cases, a rough estimation of the uncertainty in p_k can still be given as the intersection with the k axis, $[\Delta(m, \alpha)/a_{kk}]^{1/2}$.

8. ERROR ESTIMATION OF THE DERIVED PARAMETERS

All the derived parameters depend on m fit parameters (four or six times the number of velocity component). Let us call

$$p_k^0, \quad k = 1, \dots, m \quad (65)$$

the values of the fit parameters, and $\sigma(p_k)$ their errors, found from the increase in the fit residual χ^2 (see §7). Let d be any of the parameters derived from the fit parameters, $d = d(p_1, \dots, p_m)$, for instance τ_m or $N(\text{NH}_3)$. For every fit parameter p_k^0 ($k = 1, \dots, m$) we evaluate the values of the derived parameter when we increase and decrease the value of the k -th fit parameter by its error $\sigma(p_k)$,

$$\begin{aligned} d_k^+ &= d(p_1^0, \dots, p_k^0 + \sigma(p_k), \dots, p_m^0), \quad k = 1, \dots, m \\ d_k^- &= d(p_1^0, \dots, p_k^0 - \sigma(p_k), \dots, p_m^0), \quad k = 1, \dots, m \end{aligned} \quad (66)$$

Assuming that the errors of the fit parameters are statistically independent, we can estimate the error $\sigma(d)$ as

$$\sigma^2(d) = \sum_{k=1}^m \left(\frac{d_k^+ - d_k^-}{2} \right)^2. \quad (67)$$

TABLE 5
COMPARISON OF THE RESULTS OBTAINED WITH *HfS* AND CLASS FOR A SAMPLE OF $\text{NH}_3(1, 1)$ SPECTRA.

Id. ^a	ΔV (km s ⁻¹)	V_{LSR} (km s ⁻¹)	$A(1 - \exp(-\tau_m))$ (K)	$1 - \exp(-\tau_m)$	$A\tau_m$ (K)	τ_m	Rout. ^b
1	1.221 ± 0.194	-3.029 ± 0.061	6.2 ± 0.8	0.2 ± 0.5	7.1 ± 3.5	0.3 ± 0.9	<i>HfS</i>
	1.240 ± 0.091	-3.030 ± 0.027	7.1 ± 1.0	0.3 ± 0.3	C
2	0.712 ± 0.098	-4.543 ± 0.023	15.7 ± 0.9	0.996 ± 0.005	> 88	> 5.6	<i>HfS</i>
	0.750 ± 0.007	-4.550 ± 0.003	85.9 ± 1.6	5.54 ± 0.13	C
3	0.569 ± 0.066	-20.445 ± 0.030	4.5 ± 0.4	0.970 ± 0.023	16.2 ± 4.8	3.5 ± 1.1	<i>HfS</i>
	1.030 ± 0.001	-20.500 ± 0.011	8.0 ± 0.4	1.4 ± 0.2	C
4	0.199 ± 0.023	-20.382 ± 0.022	3.4 ± 0.3	1.000 ± 0.002	29 ± 19	8.4 ± 5.5	<i>HfS</i>
	1.030 ± 0.001	-20.400 ± 0.011	9.4 ± 0.4	1.26 ± 0.16	C
5	3.753 ± 0.358	-2.180 ± 0.168	6.8 ± 0.6	0.7 ± 0.2	12.3 ± 4.0	1.3 ± 0.8	<i>HfS</i>
	3.410 ± 0.039	-2.160 ± 0.022	9.87 ± 0.10	0.84 ± 0.03	C

^aSpectra analyzed in the comparison. 1: low optical depth; 2: very high optical depth; 3, 4: lines narrower than the channel width (0.6 km s⁻¹), moderate and high optical depths; 5: very broad lines.

^bRoutines being compared. *HfS*: present work; C: CLASS, using method $\text{NH}_3(1,1)$

TABLE 6
COMPARISON OF THE RESULTS OBTAINED WITH *HfS* AND THE ROUTINES OF ROSOLOWSKY ET AL. (2008) FOR A SAMPLE OF $\text{NH}_3(1, 1)$ AND (2, 2) SPECTRA.

Id. ^c	ΔV^b (km s ⁻¹)	V_{LSR} (km s ⁻¹)	$A\tau_{1m}^b$ (K)	τ_{1m}^b	T_{ex} (K)	T_k (K)	$N(\text{NH}_3)^a$ (10 ¹³ cm ⁻³)		Rout. ^d
							($f \ll 1$)	($f = 1$)	
16	0.335 ± 0.034	4.596 ± 0.013	0.90 ± 0.12	1.12 ± 0.12	3.53 ± 0.12	9.9 ± 1.9	3.8 ± 2.1	17.1 ± 9.9	<i>HfS</i>
	0.332 ± 0.014	4.614 ± 0.007	1.85 ± 0.8	< 13.	...	> 1.27	R
31.1	0.390 ± 0.008	4.618 ± 0.005	5.67 ± 0.12	1.56 ± 0.12	6.37 ± 0.06	12.6 ± 1.7	20.3 ± 3.5	35.7 ± 6.2	<i>HfS</i>
	0.400 ± 0.024	4.6 ± 0.2	...	1.875 ± 0.001	...	11.7 ± 0.1	...	23 ± 1.0	R
31.2	0.327 ± 0.023	5.984 ± 0.016	1.37 ± 0.14	1.52 ± 0.14	3.64 ± 0.09	10.6 ± 2.3	5.1 ± 2.7	21 ± 11	<i>HfS</i>
	0.306 ± 0.047	6.01 ± 0.05	...	0.870 ± 0.001	...	10.4 ± 0.1	...	10 ± 1.0	R
47	0.443 ± 0.007	8.1714 ± 0.0020	16.83 ± 0.38	4.09 ± 0.38	6.86 ± 0.04	11.49 ± 0.30	75.9 ± 3.6	126.8 ± 5.9	<i>HfS</i>
	0.444 ± 0.001	8.1840 ± 0.0007	...	4.075 ± 0.035	7.82 ± 0.02	11.69 ± 0.04	...	73.2 ± 0.7	R
89	0.294 ± 0.006	8.1617 ± 0.0024	12.00 ± 0.35	3.23 ± 0.35	6.46 ± 0.09	9.9 ± 1.0	44.6 ± 8.4	78 ± 15	<i>HfS</i>
	0.294 ± 0.002	8.1790 ± 0.0010	...	2.75 ± 0.10	7.9 ± 0.1	10.5 ± 0.1	...	39 ± 1	R
93	0.227 ± 0.005	6.002 ± 0.003	4.93 ± 0.23	2.48 ± 0.23	4.73 ± 0.09	9.9 ± 0.9	14.1 ± 2.4	33.6 ± 5.6	<i>HfS</i>
	0.231 ± 0.002	6.022 ± 0.001	...	1.65 ± 0.10	6.1 ± 0.2	10.5 ± 0.2	...	14 ± 1.0	R
95	0.343 ± 0.003	6.044 ± 0.002	21.95 ± 0.11	6.66 ± 0.11	6.04 ± 0.02	9.70 ± 0.01	98.8 ± 0.8	181.6 ± 1.2	<i>HfS</i>
	0.342 ± 0.002	6.0635 ± 0.0009	...	6.90 ± 0.10	6.75 ± 0.02	10.01 ± 0.06	...	106 ± 2.0	R

^aBeam-averaged NH_3 column densities calculated for filling factors $f \ll 1$ and $f = 1$.

^bCalculated from the values of σ_V and τ_{tot} given in Rosolowsky et al. (2008)

^cIdentification of the examples of Rosolowsky et al. (2008). 16: low optical depth, very weak (2, 2); 31: multicomponent; 47: high signal-to-noise ratio; 89: narrow lines; 93: good fit, weak (2, 2); 95: high optical depth.

^dRoutines being compared. *HfS*: present work; R: Rosolowsky et al. (2008).

9. COMPARISON OF *HfS* WITH OTHER ROUTINES

The fits obtained with *HfS* and *HfS_nh3* were compared with those obtained with other commonly used routines. In Tables 5 and 6 we show some examples of fits performed with *HfS* and CLASS for the $\text{NH}_3(1, 1)$ line, and the Rosolowsky routine (Rosolowsky et al. 2008) for the simultaneous fit of the $\text{NH}_3(1, 1)$ and (2, 2) lines. The fits shown cover cases of low and high optical depth, and narrow and wide linewidths. As can be seen in the tables, there is in general agreement between the *HfS* results and the other routines.

Regarding the comparison with CLASS, in the case of linewidths lower than the channel width (spectra 3 and 4

in Table 5), CLASS gives a fixed value for the linewidth, higher than the channel width and much higher than the actual linewidth. As a consequence the values of $A\tau_m$ and τ_m given by CLASS are scaled down by roughly the same factor, so that CLASS is consistent with *HfS* in the values of $A\tau_m\Delta V$ and $\tau_m\Delta V$. In general, the errors given by CLASS appear to be underestimated.

The comparison with the Rosolowsky routine was performed for the examples of fits given in Rosolowsky et al. (2008). The fitted parameters were taken from Table 3 of the electronic edition of Rosolowsky et al. (2008), and the raw

spectral data were retrieved from the COMPLETE Web site.⁵ For the *HfS_nh3* fits, the data were Hanning-smoothed with $\text{HFHW} = 2$ and fitted using $\text{Nksample} = 400$ (see below). For the parameters obtained specifically from the simultaneous fit of the $\text{NH}_3(1, 1)$ and $(2, 2)$ lines, there is a good agreement in the values obtained for T_k , and the values of ammonia column density reported in Rosolowsky et al. (2008) lie between the two limiting cases given by *HfS_nh3*, for filling factors $f \ll 1$ and $f = 1$.

10. DESCRIPTION OF THE *HfS* PROCEDURES

While the *HfS_nh3* procedures fit the $\text{NH}_3(1, 1)$ and $(2, 2)$ transitions simultaneously, the general *HfS* procedures fit a single transition, selected among those stored in the file *hfs_transitions.dat*. This file has to be located in the working directory, or in the directory pointed at by the environment variable *HFS_DIR* (see Appendix A). The first transition in the file is a single line at $V_{\text{LSR}} = 0$, useful for fitting a single Gaussian line. Other transitions in the file include $\text{NH}_3(1, 1)$ and $(2, 2)$, $\text{NH}_2\text{D}(1_{11}, 1_{10})$, $\text{N}_2\text{H}^+(1-0)$, CN , HCN , H^{13}CN , C_2H , C_2D , C^{17}O , and more transitions can be added easily to the file. The criterion for defining the “main component” of a transition is, in general, all the hyperfines with a velocity offset less than 0.001 km s^{-1} , and for $\text{NH}_3(1, 1)$ and $(2, 2)$, less than 0.6 km s^{-1} . Any transition not appearing in the file is assumed to be single.

An optional Hanning smoothing of the spectrum can be performed prior to fitting. You can select the Hanning filter half-width, *HFHW*. For *HFHW* = 0 no smoothing is performed. *HFHW* = 1 is the standard 3-point Hanning smoothing, and the resulting spectrum has half the initial number of channels. In general, the Hanning smoothing encompasses $2 \times \text{HFHW} + 1$ points, resulting in a final number of channels *HFHW*+1 times smaller.

The iterative process is controlled by two parameters, *Nksample*, the number of thousands of samples of the parameter space, and *Final_Range*, the ratio of ranges of the last loop and initial search ranges. The number of loops is taken as $\text{nloop} = \text{Nksample}^{1/2}$, and the number of seeds and descendants is taken as $\text{nseed} = \text{ndesc} = (\text{nloop} \times 1000)^{1/2}$, so that $\text{nloop} \times \text{nseed} \times \text{ndesc} = \text{Nksample} \times 1000$. For each loop the ranges will be decreased a factor of $f = \text{Final_Range}^{1/(\text{nloop}-1)}$.

The initial values for the fit are guessed from the intensity, position, and width of the data peak for the first component, and of the residual (data minus previous components) for the rest of components (up to a maximum of 9). The main line optical depth is set arbitrarily to 0.7 ($1 - e^{-\tau_m} = 0.5$), or to an arbitrary low value (10^{-6}) for a single Gaussian fit.

Values for the initial search ranges are calculated by the procedures. For fitting a single Gaussian line the search range of τ_m^* is made 0 to keep it constant. Additional constraints for the search ranges of the parameters are $A_m^* > 0$; $0 < \tau_m^* < 1$; and $\Delta V > \Delta V_{\text{min}}$, where ΔV_{min} is the thermal linewidth for a kinetic temperature $T_k = T_{\text{bg}}$ of a large molecule (for the general *HfS* procedures, 0.025 km s^{-1} for a mass of $200 m_{\text{H}}$), or of NH_3 (0.086 km s^{-1} for the *HfS_nh3* procedures).

Note that the final value of a fit parameter can be outside the initial range for the parameter. For an initial range $2r$, and a range decreasing factor per loop f , the final value of a

parameter can be up to roughly $r/(1 - f)$ apart from its initial value. For instance, for the default values $\text{Nksample} = 200$ (corresponding to $\text{nloop} = 14$), and $\text{Final_Range} = 0.05$, we have $f \simeq 0.8$ and $r/(1 - f) \simeq 5r$.

The different procedures that compose *HfS* are described in the following.

10.1. *hfs_fit*, *hfs_nh3*

These are interactive graphic procedures for fitting simultaneously multiple velocity components of an spectral line with hyperfine structure, or to a pair of $\text{NH}_3(1, 1)$ and $(2, 2)$, to spectra read from data files, and generating files with the synthetic spectra. See an example of a fitting run of *hfs_fit* in Appendix D and of *hfs_nh3* in Appendix E.

You can set any number of velocity components, for which the procedures propose a first guess. Alternatively, you can use the cursor to add or delete components. The position of a new component is set at the cursor position, its intensity is the intensity at the cursor position, and the FWHM is estimated around the cursor position. The values of the initial search ranges can be changed. Any of the parameters can be kept constant by setting its range to 0.

INPUT

The spectra to fit are read from ASCII files with a pair of values (velocity, intensity) per line. Lines beginning with “!” or “#” are ignored. The file(s) can be given as argument(s) to *hfs_fit* and *hfs_nh3*:

```
$ hfs_fit <source>.dat
$ hfs_nh3 <source_11>.dat <source_22>.dat
```

OUTPUT

- *hfs_fit.log* or *hfs_nh3.log*, log file with the details of the fitting session.
- *<source>.synt*, or *<source_11>.synt* and *<source_22>.synt*, ASCII files with the synthesized spectra. The files have a header (lines beginning with “!”) with the values of the parameters of the fitted spectra for each velocity component. The files are readable by GREG of the package GILDAS. The file is overwritten for every new fit. See an example in Appendix G.
- *<source>.eps*, plot showing the data, the components fitted, and the residual. See some examples in Fig. 2. The file is overwritten for every new fit.

10.2. *hfs_file*

This is a batch procedure to fit the hyperfine structure of the same transition (and a single velocity component) for a set of data files, and generating a list of files with the synthetic spectra. The fit procedure is the same as that of *hfs_fit*, but it is not interactive.

INPUT

The input is a parameter file that can be given as argument when running the procedure:

```
$ hfs_file <file_list>.par
```

The contents of the file is as follows:

⁵ https://www.cfa.harvard.edu/COMPLETE/data_html_pages/GBT_NH3.html

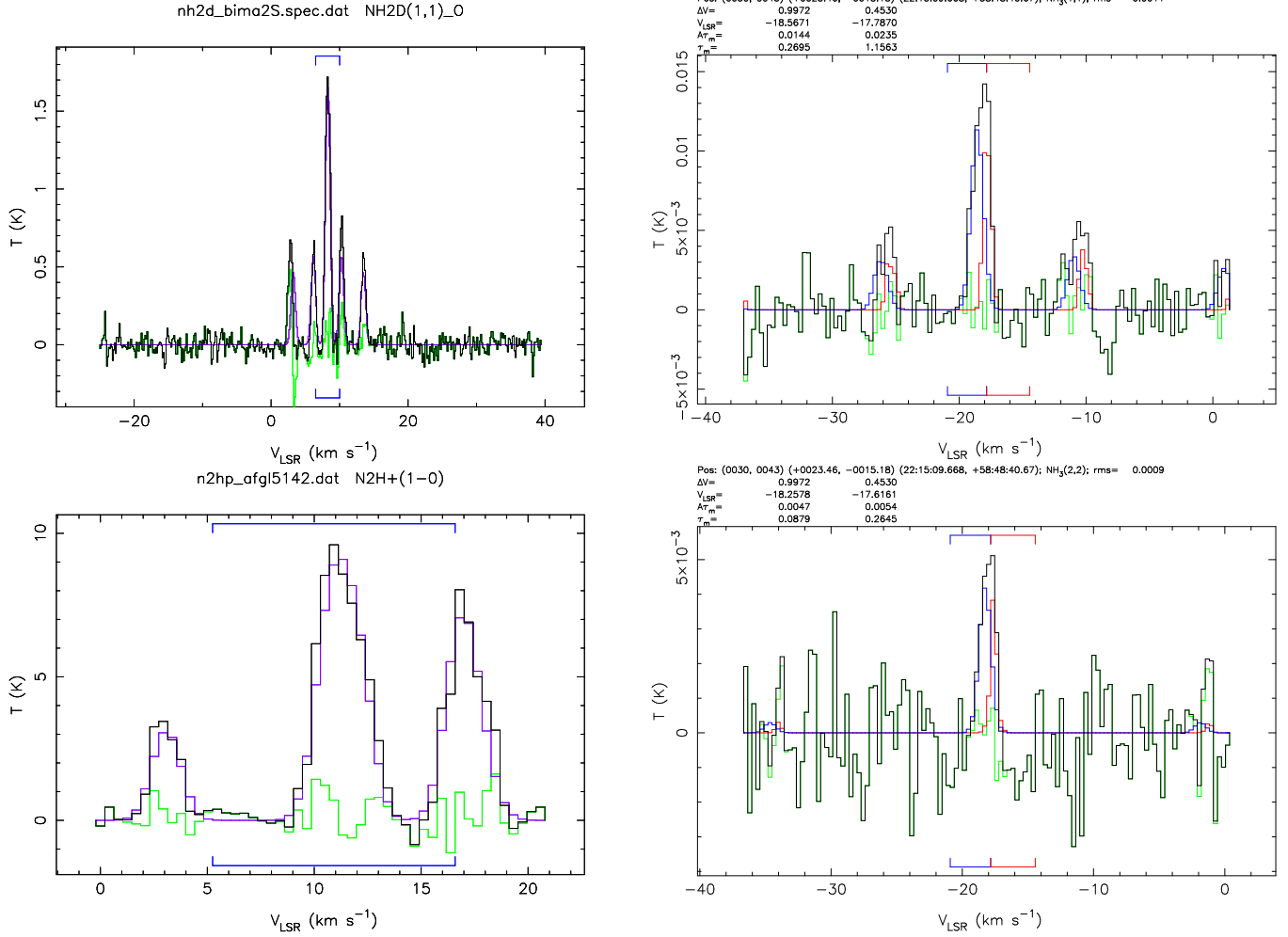


FIG. 2.— Left: $\text{NH}_2\text{D}(1,1)$ (top) and $\text{N}_2\text{H}^+(1-0)$ (bottom) fits using `hfs_fit`. The color lines indicate the data (black), fit (magenta), and residual (green). Right: $\text{NH}_3(1,1)$ and $(2,2)$ simultaneous fit of two velocity components using `hfs_nh3`. The blue and red lines indicate the two velocity components fitted. The search range for each component is indicated by the blue and red segments at the top and bottom of each plot.

1. Transition name, e.g. " $\text{NH}_3(1,1)$ ".
2. Iteration parameters: `Nksample`, `Final_Range`.
3. and following lines: list of files, one file per line

See an example of `<file_list>.par` in Appendix F.

OUTPUT

- `<file_list>.log`, log file with the details of the fitting process for all the files in `<file_list>.par`.
- `<file_list>.out`, ASCII file with the values and uncertainties of the parameters fitted for each file of the list. The first lines beginning with “!” are the header and give information about the transition and the column headers.
- `<file_list>.ps`, PostScript file with plots of the data, the components fitted, and the residual for all the files in the list.
- `<file_#>.synt`, an ASCII file for each file in `<file_list>.par`, with the synthesized spectrum. The file has a header (lines beginning with “!”) with the values of the parameters of the fitted spectrum.

10.3. `hfs_cube_sp`, `hfs_nh3_cube_sp`

These are single processor batch procedures for fitting simultaneously multiple velocity components of spectra from 3-axes FITS data cubes. A subimage of the FITS data cube can be selected, and optional boxcar averaging of pixels and Hanning filtering of the spectra can be performed. The procedures are similar to `hfs_fit` and `hfs_nh3`, but they are not interactive. The maximum dimension of the image and the maximum number of channels of the data cube is arbitrary, i.e. the only limitation is the amount of memory available by the computer.

INPUT

The input is a parameter file that has to be given as argument when running the procedures:

```
$ hfs_cube_sp <parameter>.par
$ hfs_nh3_cube_sp <parameter>.par
```

The parameter file for `hfs_cube_sp` has 10 lines (see an example in Appendix F):

1. Transition name, e.g. " $\text{C}170(1-0)$ ".
2. FITS data cube file to read. The file must have 3 non-degenerate axes, in this order: x position, y position,

velocity channel.

3. Rms of channels without emission, minimum SNR of spectra to be fitted. Components with a peak intensity below SNR times rms are not fitted.
4. Number of velocity components to fit, *Ncomp*, between 1 and 9.
5. Range of channels for each component: $2 \times Ncomp$ values, with the first and last channel of the velocity range for each component. The channel ranges must be non overlapping. For the first component, 0 defaults to 1 (first), *nchan* (last).
6. Hanning filter half-width (channels), 0 for no filtering.
7. Boxcar smoothing radius (pixels), 0 for no smoothing.
8. Subimage to be fitted: first X pixel, last X pixel, X increment. 0 defaults to 1 (first), *ndim1* (last), 1 (increment).
9. Subimage to be fitted: first Y pixel, last Y pixel, Y increment. 0 defaults to 1 (first), *ndim2* (last), 1 (increment).
10. Iteration parameters: *Nksample*, *Final_Range*.

The parameter file for *hfs_nh3_cube_sp* has 11 lines (see an example in Appendix F):

1. (1, 1) FITS data cube file to read. The file must have 3 non-degenerate axes, in this order: x position, y position, velocity channel.
2. (1, 1) rms of channels without emission, minimum SNR of spectra to be fitted. Components with a peak intensity below SNR times rms are not fitted.
3. Number of velocity components to fit, *Ncomp*, between 1 and 9.
4. (1, 1) range of channels for each component: $2 \times Ncomp$ values, with the first and last channel of the velocity range for each component. The channel ranges must be non overlapping. For the first component, 0 defaults to 1 (first), *nchan* (last).
5. (2, 2) FITS data cube file to read. The file must have the same geometry as that of the (1, 1) FITS file.
6. (2, 2) rms of channels without emission, minimum SNR of spectra to be fitted. Components with a peak intensity below SNR times rms are not fitted.

The 5 following lines are the same as for the case of *hfs_cube_sp*.

OUTPUT

- *log/<parfile>.log*, log file in folder *log* with the details of the fitting process for all the pixels of the subimage.

- *<parfile>.comp#.out*, an ASCII file for each velocity component with the values of the parameters fitted and the line and physical parameters for each pixel of the subimage, and their uncertainty. The first lines beginning with "!" are the header and give information about the parameter file, FITS files, velocity component number, velocity range of the component, Hanning filtering applied, smoothing boxcar radius, and column headers. The files are used by *hfs_view*, and are readable by GREG. See an example in Appendix G.
- *ps/<parfile>.<xoffset>.ps*, PostScript files in folder *ps*, with plots of the data, the components fitted, and the residual for all pixels of the subimage with a given *<xoffset>* in arcsec.
- *maps/<parfile>.<parameter>.comp#.fits*, FITS files in folder *maps*, with maps, for each velocity component, of the parameters fitted and the line and physical parameters, and their uncertainty. Each FITS file has two planes, the first plane with the values of the parameter, and the second plane with the uncertainties. See in Fig. 3 an example of the the maps obtained from a 2-velocity-components fit of the $C^{18}O$ line for a FITS data cube.

10.4. *hfs_cube_mp*, *hfs_nh3_cube_mp*

These are multiprocessor procedures that use Open MPI (see Appendix A) and run in parallel using a number of processors available in the machine, or in more than one host. The multiprocessor procedures are naked versions of the single-processor versions, *hfs_cube_sp* and *hfs_nh3_cube_sp*, without any graphic output. The instructions for running these procedures can be found in Appendix C.

10.5. *hfs_view*, *hfs_nh3_view*

These are interactive graphic procedures for displaying spectra from FITS data cubes and the corresponding synthetic spectra fitted with *hfs_cube* or *hfs_nh3_cube*. A plot of the integrated intensity is shown, and you can select with the mouse the position for which the the spectrum (data and synthetic) is shown. You can select to show the integrated intensity for all channels, or for the channel ranges of each velocity component. The data and synthetic spectra at any pixel can be extracted in ASCII files.

INPUT

The input is the same parameter file used as input to *hfs_cube* or *hfs_nh3_cube*. From the information in this file, *hfs_view* reads the corresponding FITS data cubes and the *<parfile>.comp#.out* file for each velocity component (created by *hfs_cube* or *hfs_nh3_cube*), with the parameters of the fitted spectra.

OUTPUT

For any pixel you can extract the data and synthetic spectra:

- *<parfile>.<xoffset>.<yoffset>.spec*, or *<parfile>.11.<xoffset>.<yoffset>.spec* and *<parfile>.22.<xoffset>.<yoffset>.spec*, data spectra for the pixel selected, with the Hanning filtering and boxcar smoothing given in the *<parfile>.par* file. See an example in Appendix G.

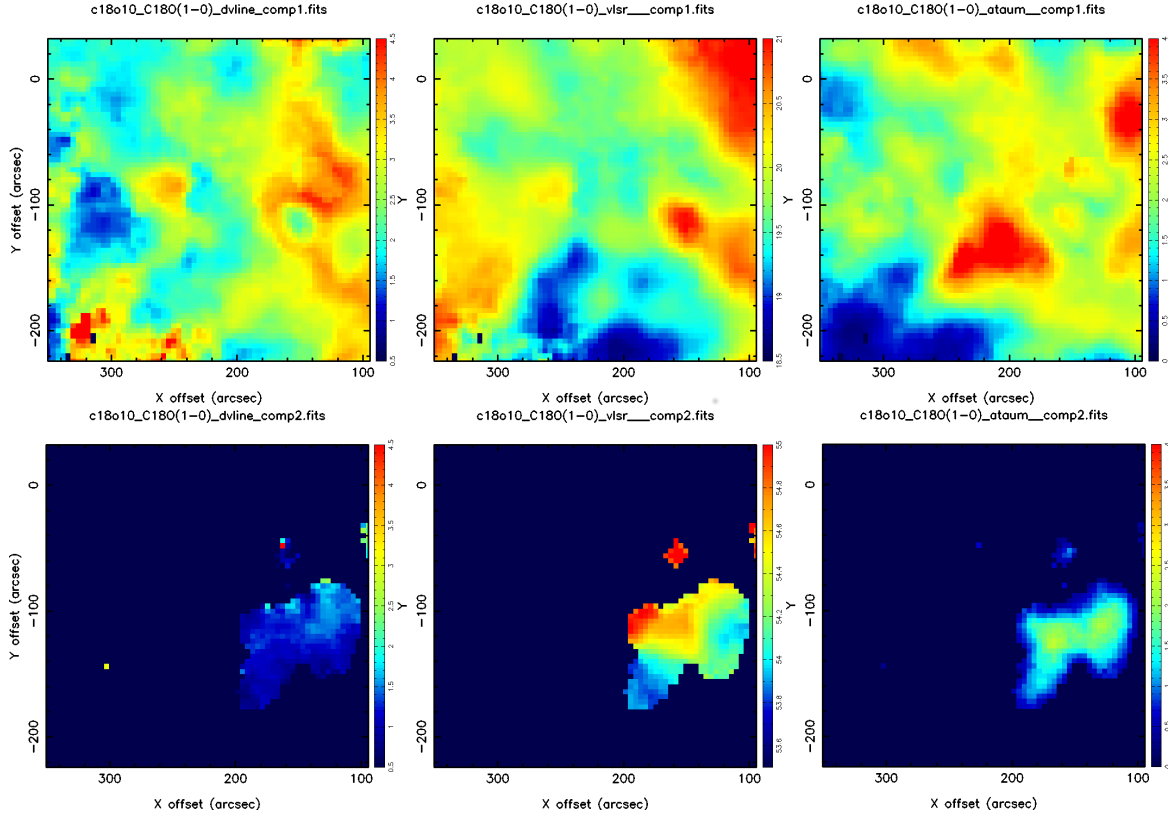


FIG. 3.— Example of maps of the parameters fitted with `hfs_cube` of two velocity components to the C^{18}O (1–0) transition for a 64×64 data cube. *Top*: component 1, *bottom*: component 2, *left*: ΔV , *middle*: V_{LSR} , *right*: $A\tau_m$.

- `<parfile>.<xoffset>.<yoffset>.synt`, or
`<parfile>_11.<xoffset>.<yoffset>.synt` and
`<parfile>_22.<xoffset>.<yoffset>.synt`, syn-
 thetic spectra fitted for the pixel selected.

10.6. `hfs.blanking`

This is an auxiliary procedure to flag the output FITS files of `hfs_cube` and `hfs_nh3_cube`, according to the parameter values, errors, or SNR. The FITS files have two planes, the first one with the values of the parameter at each pixel, and the second one with the error of the value. The procedure reads the values and errors in the FITS files, and allows you to blank the pixels that fulfill a criterion on parameter error, value, or value/error, above or below a cutoff value.

INPUT

`<parfile>.<parameter>.comp#.fits`, any FITS file created by `hfs_cube` or `hfs_nh3_cube`, for each velocity component and parameter.

OUTPUT

`<parfile>.<parameter>.comp#.blank.fits`, output 2-axes FITS file, with the values of the parameter for non-flagged pixels, and NaN for flagged-out pixels.

10.7. `hfs.synt`

Auxiliary procedure to create a synthetic spectrum, with the option of adding Gaussian noise. You can select the transition, number of channels, spectral resolution, line parameters, and noise level.

OUTPUT

- `hfs.synt.synt`, ASCII file with the synthetic spectrum generated. The file has a header (lines beginning with “!”) with the values of the parameters used for generating the synthetic spectrum.
- `hfs.synt.eps`, plot of the synthetic spectrum.

The author thanks Pau Estalella for suggesting the use of pseudo-random sequences, Ferran Sala for helpful discussions on the projections of a quadric, and Salvador Curiel for helping with the implementation of the multiprocessor procedures. Thanks also to Álvaro Sánchez-Monge for reading the manuscript and, together with Aina Palau, Gemma Busquet and Carmen Juárez, for testing *HfS*, finding bugs, and suggesting improvements. This work has been partially supported by the Spanish MINECO grant AYA2014-57369-C3 (cofunded with FEDER funds) and MDM-2014-0369 of IC-CUB (Unidad de Excelencia ‘María de Maeztu’).

REFERENCES

- Anglada, G., Estalella, R., Mauersberger, R., Torrelles, J. M., Rodríguez, L. F., Cantó, J., Ho, P. T. P., D’Alessio, P. 1995, *ApJ*, 443, 682
 Avni, Y. 1976, *ApJ*, 210, 642
 Cantó, J., Curiel, S., Martínez-Gómez, E. 2009, *A&A*, 501, 1259
 Danby, G., Flower, D. R., Valiron, P., Schilke, P., & Walmsley, C. M. 1988 *MNRAS*, 235, 229
 Estalella, R. 2016, *HfS: Hyperfine Structure fitting tool*, Astrophysics Source Code Library, record ascl:1607.011

- Estalella, R., Anglada, G. 1997, "Introducción a la Física del Medio Interestelar", Colección Textos Docents, n. 50, 2nd edition: Edicions de la Universitat de Barcelona, Spain
- Estalella, R., López, R., Anglada, G., Gómez, G., Riera, A., Carrasco-González, C. 2012, *AJ*, 144, 61
- Faure, A., Hily-Blant, P., Le Gal, R., Rist, C., Pineau des Forêts, G. 2013, *ApJ*, 770, L2
- Goddi, C., Zhang, Q., Moscadelli, L. 2015, *A&A*, 573, A108
- Halton, J. H. 1964, *Commun. ACM*, 7, 701
- Ho, P. T. P., Townes, C. H. 1983, *ARAA*, 21, 239
- Kukolich, S. G. 1967, *Phys. Rev.*, 156, 83
- Mangum, J. G., Shirley, Y. L. 2015, *PASP*, 127, 266
- Maret, S., Faure, A., Scifoni, E., & Wiesenfeld, L. 2009, *MNRAS*, 399, 425
- McConnell, A. J. 2011, "Applications of Tensor Analysis", Dover Publications
- Osorio, M., Anglada, G., Lizano, S., D'Alessio, P. 2009, *ApJ*, 694, 29
- Palau, A., Estalella, R., Girart, J. M., Fuente, A., Fontani, F., Commerçon, B., Busquet, G., Bontemps, S., Sánchez-Monge, Á., Zapata, L. A., Zhang, Q., Hennebelle, P., di Francesco, J. 2014, *ApJ*, 785, 42
- Poynter, R. L., Kakar, R. K. 1975, *ApJ*, 29, 87
- Rosolowsky, E. W., Pineda, J. E., Foster, J. B., Borkin, M. A., Kauffmann, J., Caselli, P., Myers, P. C., Goodman, A. A. 2008, *ApJSS*, 175, 509
- Sánchez-Monge, Á., Palau, A., Fontani, F., Busquet, G., Juárez, C., Estalella, R., Tan, J. C., Sepúlveda, I., Ho, P. T. P., Zhang, Q., Kurtz S. 2013, *MNRAS*, 432, 3288
- Sepúlveda, I., Anglada, G., Estalella, R., López, R., Girart, J.M., Yang, J. 2011, *A&A*, 527, A41
- Svoboda, B. E., Shirley, Y. L., Battersby, C., Rosolowsky, E. W., Ginsburg, A. G., Ellsworth-Bowers, T. P., Pestalozzi, M. R., Dunham, M. K., Evans, N. J., II, Bally, J., Glenn, J. 2016, *ApJ*, 822, 59
- Sobol, I. 1967, *USSR Computational Mathematics and Mathematical Physics*, 7, 86
- Swift, J. J., Welch, W. J., & Di Francesco, J. 2005 *ApJ*, 620, 823
- Walmsley, C. M. & Ungerechts, H., 1983. *A&A*, 122, 164
- Wall, J. V. & Jenkins, C. R. 2003, "Practical Statistics for Astronomers", Cambridge University Press

APPENDIX

A. *HfS* REQUISITES AND INSTALLATION

The *HfS* procedures run on a Linux or Mac OS X system with a Fortran 90 compiler (for instance *gfortran*), and use the PGplot Graphics Subroutine Library compiled with *gfortran* (see Appendix B), for the graphic output, and Open MPI for the multiprocessor procedures *hfs_cube.mp* and *hfs_nh3_cube.mp*.

HfS can be freely downloaded as a file *hfs.tgz* from the Astrophysics Source Code Library, record ascl:1607.011⁶ Once you have downloaded *hfs.tgz*, untar the file in your installation directory (as root or using *sudo*), for instance */usr/local/hfs*,

```
$ mkdir /usr/local/hfs
$ mv hfs.tgz /usr/local/hfs
$ cd /usr/local/hfs
$ tar -xzf hfs.tgz
```

If necessary, edit the first lines of the shell script *hfs_compile* to change the lines

```
compiler="gfortran"
libraries="-lpgplot -lX11"
```

You may need to give the location of the PGplot library with the option

```
libraries="-lpgplot -lX11 -L/usr/local/lib/pgplot"
```

Mac OS X users may need to indicate the location of the X11 library too

```
libraries="-lpgplot -lX11 -L/usr/local/lib/pgplot -L/usr/X11/lib"
```

Run the compile script

```
$ ./hfs_compile
```

The shell script *hfs.links* create symbolic links in *\$exe_dir* pointing at the *HfS* procedures. If necessary, change the line

```
exe_dir="/usr/local/bin"
```

and run the links script

```
$ ./hfs_links
```

Define the environment variable *HFS_DIR*, pointing at your installation directory. It is used by *HfS* to find the files *hfs_transitions.dat* and *hfs_fit.help*. This can be done by adding to your *.bashrc* or *.bash_profile* the lines

```
# hfs
export HFS_DIR=/usr/local/hfs
```

⁶ <http://ascl.net/1607.011>

B. INSTALLATION OF PGPLOT WITH GFORTRAN

Once you have downloaded the PGplot distribution file, follow the normal installation procedure (as root or using `sudo`):

```
$ mv pgplot5.2.tar.gz /usr/local/src
$ cd /usr/local/src
$ tar xzvf pgplot5.2.tar.gz
$ mkdir /usr/local/pgplot
$ cd /usr/local/pgplot
$ cp /usr/local/src/pgplot/drivers.list .
```

Edit the file `drivers.list` and uncomment (select) the drivers

`/NULL` (null device),
`/PS`, `/VPS`, `/CPS`, `/VCPS` (PostScript drivers), and
`/XWINDOW`, `/XSERVE` (X window drivers).

Run `makemake` to prepare the makefile for a Linux system with `g77_gcc` compiler,

```
$ /usr/local/src/pgplot/makemake /usr/local/src/pgplot linux g77_gcc
```

Edit the file `makefile` and replace lines 25 and 26:

```
FCOMPL=g77
FFLAGC=-u -Wall -fPIC -O
```

by the following lines:

```
FCOMPL=gfortran
FFLAGC=-ffixed-form -ffixed-line-length-none -u -Wall -fPIC -O
```

Continue the normal installation procedure,

```
$ make
$ make clean
```

and, assuming that `LD_LIBRARY_PATH` points at `/usr/local/lib`,

```
$ cd /usr/local/lib
$ ln -s /usr/local/pgplot/libpgplot.so .
```

Define the environment variable `PGPLOT_DIR`, pointing at your installation directory. This can be done by adding to your `.bashrc` or `.bash_profile` file the lines

```
# PGplot
export PGPLOT_DIR='/usr/local/pgplot'
```

Additionally, you can customize PGplot by adding the following lines to your `.bashrc` or `.bash_profile`:

```
# Default xwindow device
export PGPLOT_DEV='/xwin'
# Default white background
export PGPLOT_BACKGROUND='white'
# Default black foreground
export PGPLOT_FOREGROUND='black'
# Marking text written in the ps file so it can be edited
export PGPLOT_PS_VERBOSE_TEXT='yes'
# Starting pgxwin_server with 256 colors, server window not visible
/usr/local/pgplot/pgxwin_server -win_maxColors 256 -server_visible False
```

C. RUNNING THE MULTIPROCESSOR PROCEDURES HFS_CUBE_MP AND HFS_NH3_CUBE_MP

You need to have Open MPI installed to be able to run `hfs_cube_mp` or `hfs_nh3_cube_mp`. To run the procedures, type

```
$ mpirun -np <N> hfs_cube_mp <parfile>.par
$ mpirun -np <N> hfs_nh3_cube_mp <parfile>.par
```

where `<N>` is the number of processors to use. To know the number (and characteristics) of processors in a Linux system, you can type

```
$ cat /proc/cpuinfo
```

For running in more than one host, for example, in localhost and another host <otherhost>, type

```
$ mpirun -np <N> -host localhost,<otherhost> hfs_cube_mp <parfile>.par
$ mpirun -np <N> -host localhost,<otherhost> hfs_nh3_cube_mp <parfile>.par
```

Here <N> is the total number of processors to use, distributed among the hosts listed after -host. If you want to know which are the processes run in each host, add the option -display-map.

Requisites for running in a remote host <otherhost>:

- ssh access to <otherhost>, without having to enter the password, i.e. with your local id_rsa.pub added to <otherhost>: .ssh/authorized_keys2.
- Open MPI installed in <otherhost>. Your PATH and LD_LIBRARY_PATH in <otherhost> have to point at the openmpi bin and lib folders in the installation directory. For example, if openmpi is installed in /usr/local/openmpi, you can include in the file <otherhost>: .bashrc the lines

```
export PATH=$PATH:/usr/local/openmpi/bin
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/openmpi/lib
```

- The same data file structure in <otherhost> and in localhost, i.e. the same directory from where you run hfs_cube_mp or hfs_nh3_cube_mp, with the same parameter file <parfile>.par and data files. The log files of the different processors, log/<parfile>_##.log, will be written in the host where each processor runs.

D. EXAMPLE OF A HFS_FIT RUN

```
$ hfs_fit nh311_thin.dat
```

```
HfS_fit. HyperFine Spectra multicomponent fit
Robert Estalella, 2015/06
```

```
***** Use a terminal with at least 94 columns *****
Date: 2016/03/22 Time: 12:45:04
```

```
Data file: nh311_thin.dat
N. of data points read:      87
Reference channel (V=0):    59.6269
Channel width (km s-1):      0.3089
Off-line rms:                0.2104
Transition:                  NH3(1,1)
tau_tot/tau_m:               2.0000
```

```
Fit rms:      0.3933
Present values and search ranges (ncomp= 1)
Param:      Delta_V      V_lsr A(1-e-t-tau_m) 1-e-tau_m
Comp:   1  -----
Value:      1.2103      -2.9734      6.1716      0.5000
Range:      1.2103      1.2103      0.3841      0.5000
```

```
HfS fit menu-----
0. Quit
1. Help
2. Read data file:   nh311_thin.dat
3. Select transition: NH3(1,1)
4. Hanning smoothing
5. Select plot (1:data +2:comp +4:res +8:synt):  7
6. Change Nksample, Final_Range:      200      0.050
7. Change Ncomp= 1 and make initial guess
8. Change initial values or ranges
9. Fit 1 component(s) and estimate errors
Choose option (0-9): 9
```

```
Iteration parameters read
Nksample:      200
Final_Range:    0.050
Iteration parameters used
Nseed:         118
Ndesc:         118
Nloop:         14
Range_Fact:    0.794
```

Fitting ncomp= 1 component(s)

Loop	Comp	Delta_V	V_lsr	A(1-e^-t_m)	1-e^-tau_m	rms
0	1	1.2103	-2.9734	6.1716	0.5000	0.3933
1	1	2.1267	-3.9476	6.3392	0.9930	0.3933
2	1	1.4883	-2.9911	5.8081	0.3219	0.3784
3	1	1.1926	-3.1046	6.1852	0.2803	0.3766
4	1	1.2946	-2.9713	6.0164	0.1679	0.3762
5	1	1.2304	-3.0472	5.9933	0.1664	0.3762
6	1	1.1915	-2.9995	6.3971	0.2483	0.3762
7	1	1.2597	-3.0325	5.9275	0.2578	0.3762
8	1	1.1783	-3.0519	6.2819	0.3125	0.3762
9	1	1.1862	-3.0152	6.2391	0.3103	0.3761
10	1	1.2051	-3.0217	6.3489	0.1883	0.3761
11	1	1.2628	-3.0253	6.1733	0.1857	0.3760
12	1	1.2043	-3.0294	6.2455	0.2270	0.3760
13	1	1.2196	-3.0238	6.1633	0.2602	0.3760
14	1	1.2284	-3.0299	6.1945	0.2700	0.3760

Error estimation in progress

Fit rms: 0.3760

N. fitted par: 4

Target rms: 0.3865

Single parameters.....

Pairs of parameters.....

Par	Intersect	Projection
1	0.1105	0.1938
2	0.0585	0.0608
3	0.4400	0.7270
4	0.2665	0.5207

Derived parameters

Done

Fit rms: 0.3760

Best fit and errors

Param:	Delta_V	V_lsr	A(1-e^-t_m)	1-e^-tau_m	Derived parameters		
Comp: 1					Atau_m	tau_m	A
Value:	1.2188	-3.0303	6.2258	0.2479	7.1547	0.2849	2.5114E+01
Error:	0.1938	0.0608	0.7270	0.5207	3.4903	0.8576	4.4023E+06

E. EXAMPLE OF A HFS_NH3 RUN

\$ hfs_nh3 nh3_11.dat nh3_22.dat

NH3 (1,1) and (2,2) multicomponent fit and analysis
 Robert Estalella. 2015/12

***** Use a terminal with at least 118 columns *****

Date: 2016/03/22 Time: 12:04:18

Data file 1: nh3_11.dat

N. of data points read: 413

Reference channel (V=0): 208.0598

Channel width (km s^-1): 0.2059

Data file 2: nh3_22.dat

N. of data points read: 413

Reference channel (V=0): 207.9984

Channel width (km s^-1): 0.2057

Fit rms 1,2: 0.0582 0.0157

Present values and search ranges (Ncomp= 1)

(J,K):	(1,1)&(2,2)	(1,1)	(1,1)	(1,1)	(2,2)	(2,2)
Param:	Delta_V	V_lsr	A(1-e^-t_m)	1-e^-tau_m	V_lsr	A(1-e^-t_m)
Comp: 1						
Value:	0.9006	0.3998	1.5594	0.5000	0.4120	0.2892
Range:	0.9006	1.0000	0.0580	0.5000	1.0000	0.0157

Hfs NH3(1,1)&(2,2) fit menu_____

0. Quit


```

1. Help
2. Read data files: nh3_11.dat, nh3_22.dat
3. Physical parameter estimation
4. Hanning smoothing
5. Select plot (1:data +2:comp +4:res +8:synt): 15
6. Change Nksample, Final_Range:      200    0.050
7. Change Ncomp= 1 and make initial guess
8. Change initial values or ranges
9. Fit 1 component(s) and estimate errors
Choose option (0-9): 9

```

```

Iteration parameters read
Nksample:      200
Final_Range:   0.050
Iteration parameters used
Nseed:         118
Ndesc:         118
Nloop:         14
Range_Fact:    0.794

```

Fitting ncomp= 1 component(s)

Loop	Comp	Delta_V	V_lsr1	A(1-e ^{-t1m})	1-e ^{-tau1m}	V_lsr2	A(1-e ^{-t2m})	rms_tot
0	1	0.9006	0.3998	1.5594	0.5000	0.4120	0.2892	0.0603
1	1	1.0775	-0.4231	1.5179	0.9924	1.3918	0.3027	0.0543
2	1	0.8312	0.4102	1.5944	0.6611	1.3326	0.2791	0.0351
3	1	0.8878	0.3940	1.6405	0.8201	0.5550	0.2785	0.0310
4	1	0.7420	0.4763	1.6033	0.8780	0.0978	0.3075	0.0310
5	1	0.5446	0.3985	1.5672	0.9798	1.9346	0.3106	0.0305
6	1	0.6609	0.4237	1.5769	0.9845	0.3772	0.2785	0.0305
7	1	0.5775	0.3836	1.5560	0.9914	0.5027	0.2989	0.0294
8	1	0.5718	0.4494	1.7124	0.9730	0.2922	0.2766	0.0287
9	1	0.6237	0.4140	1.5783	0.9651	0.4681	0.3009	0.0285
10	1	0.5559	0.4449	1.4869	0.9923	0.2012	0.3003	0.0285
11	1	0.5236	0.4082	1.5732	0.9881	0.4519	0.2765	0.0285
12	1	0.5844	0.4423	1.5001	0.9866	0.4943	0.3013	0.0283
13	1	0.5683	0.4253	1.5108	0.9860	0.2885	0.2976	0.0282
14	1	0.5828	0.4155	1.4899	0.9862	0.4128	0.3338	0.0282

Error estimation in progress

```

Fit rms:      0.0282
N. fitted par: 6
Target rms:   0.0283
Single parameters.....

```

Pairs of parameters.....

Par	Intersect	Projection
1	0.0144	0.0361
2	0.0094	0.0099
3	0.0318	0.0000
4	0.0017	0.0000
5	0.0754	0.0754
6	0.0696	0.0725

Derived parameters
Done

Fit rms 1,2: 0.0232 0.0164

Best fit and errors

(J,K):	(1,1)&(2,2)	(1,1)	(1,1)	(1,1)	(2,2)	(2,2)
Param:	Delta_V	V_lsr	A(1-e ^{-t_m})	1-e ^{-tau_m}	V_lsr	A(1-e ^{-t_m})
Comp:	1					
Value:	0.5560	0.4218	1.4801	0.9910	0.4158	0.3228
Error:	0.0361	0.0099	0.0318	0.0017	0.0754	0.0725

Derived parameters

(J,K):	(1,1)	(1,1)	(2,2)	(2,2)	(1,1)&(2,2)
Param:	Atau_m	tau_m	Atau_m	tau_m	A
Comp:	1				
Value:	7.0371	4.7118	0.3638	0.2436	1.4935E+00
Error:	0.3068	0.1865	0.0927	0.0623	3.2230E-02

Physical parameters

Param:	(f=1)T _{ex} (K)	T _{rot} (K)	T _k (K)	(f<1)N(11) (cm ⁻²)	(f=1)N(11) (cm ⁻²)	(f<1)N(22) (cm ⁻²)	(f=1)N(22) (cm ⁻²)	(f<1)N(NH3) (cm ⁻²)	(f=1)N(NH3) (cm ⁻²)
Comp:	1								
Value:	4.2276	9.6960	9.9724	1.0236E+14	2.9150E+14	2.4889E+12	7.0877E+12	4.5732E+14	1.3023E+15
Error:	0.0324	0.5968	0.6588	8.0026E+12	2.2179E+13	6.5442E+11	1.8673E+12	6.5753E+13	1.8463E+14

F. EXAMPLES OF Hfs INPUT FILES

<file_list>.par

```
"NH3(1,1)"          ! Transition
400 0.05             ! Nksample, Final_Range
00117+6412.nh311.clump1.spt ! 1st data file
AFGL5142.nh311.clump1.dat  ! 2nd data file
CepA.nh311.clump1.spt     ! ...
ON1.nh311.clump1.spt      ! Last data file
```

Parameter file for hfs_cube.sp and hfs_cube.mp

```
"N2H+(1-0)"          ! Transition
"n2hp-merged-lmv-clean.fits" ! Input data cube file
0.06 8.0              ! Rms, minimum SNR to analyze the spectrum
1                      ! Number of components
104 108               ! Range of channels to search for peak
0                      ! Hanning filter half-width (chan): 0=no; >0=yes
0                      ! Boxcar smoothing radius (pixels): 0=no; >0=yes
20 236 4              ! Xpix_ini, Xpix_fin, Xpix_inc
20 236 4              ! Ypix_ini, Ypix_fin, Ypix_inc
400 0.05              ! Nksample, Final_Range
```

Parameter file for hfs_nh3_cube.sp and hfs_nh3_cube.mp

```
"l1287_11_Kv.fits"    ! (1,1) input NH3(1,1) data cube file
0.30 4.0              ! (1,1) rms, minimum SNR to analyse the spectrum
3                      ! Number of components
27 29 30 32 33 35     ! (1,1) range of channels to search for peak
"l1287_22_Kv.fits"    ! (2,2) input NH3(2,2) data cube file
0.30 4.0              ! (2,2) rms, minimum SNR to analyse the spectrum
0                      ! Hanning filter half-width (chan): 0=no; >0=yes
1                      ! Boxcar smoothing radius (pixels): 0=no; >0=yes
128 384 3             ! (1,1)&(2,2) Xpix_ini, Xpix_fin, Xpix_inc
128 384 3             ! (1,1)&(2,2) Ypix_ini, Ypix_fin, Ypix_inc
400 0.05              ! Nksample, Final_Range
```

G. EXAMPLES OF Hfs OUTPUT FILES

Header and first lines of <source>.synt

```
!Created by Hfs_nh3
!DATE      = 2016/01/12
!TIME      = 12:59:00
!TRANSITION = NH3(2,2)
!NCHAN     =          120
!DVCHAN    =       0.30851
!NCOMP     =           2
!DVLINE__1 =       1.05310
!VLSR____1 =      -18.38200
!A*TAU_M__1 =       0.00226
!TAU_M____1 =       0.13445
!DVLINE__2 =       0.00000
!VLSR____2 =       0.00000
!A*TAU_M__2 =       0.00000
!TAU_M____2 =       0.00000
!  VELOCITY  SYNTHETIC    COMP_1    COMP_2
-36.50216    0.00002    0.00002    0.00000
-36.19365    0.00006    0.00006    0.00000
-35.88514    0.00011    0.00011    0.00000
-35.57662    0.00014    0.00014    0.00000
-35.26811    0.00011    0.00011    0.00000
-34.95960    0.00006    0.00006    0.00000
-34.65109    0.00002    0.00002    0.00000
-34.34258    0.00000    0.00000    0.00000
```

Header and first lines of <source>_<xoffset>_<yoffset>.spec

```
!Created by HfS_view
!DATE          = 2016/01/12
!TIME          = 12:59:00
!FITS_FILE     = test1.fits
!NCHAN         =          124
!DVCHAN        =          0.30888
!HANNING_HWDTH =          0
!BOXCAR_RADIUS =          0
!X_PIXEL       =          30
!Y_PIXEL       =          40
!  VELOCITY    INTENSITY
-36.83265      -0.00740
-36.52377      -0.00101
-36.21490       0.00281
-35.90602       0.00570
-35.59714      -0.00071
-35.28827       0.00164
-34.97939      -0.00221
-34.67051      -0.00472
-34.36163       0.00010
-34.05275      -0.00056
-33.74388      -0.00011
-33.43500      -0.00030
-33.12612       0.00054
```

<parfile>_comp#.out

```
!Created by HfS_cube
!DATE          = 2016/01/12
!TIME          = 13:20:56
!PAR_FILE      = test2_cube.par
!FITS_FILE     = test1.fits
!TRANSITION    = NH3(1,1)
!TAU_TOT/TAU_M = 2.0000
!NCOMP         = 2
!COMPONENT     = 1
!VELOCITY_RANGE_MIN = -23.3965
!VELOCITY_RANGE_MAX = -18.7633
!HANNING_HALF_WIDTH = 0
!BOXCAR_RADIUS = 10
!  DELTA_V      ERROR      V_LSR      ERROR      A*TAU_M      ERROR      TAU_M      ERROR      RMS      XOFFSET      YOFFSET      XPIX      YPIX
8.9086E-01  1.3444E-01 -1.9046E+01  9.7873E-02  1.9160E-02  4.7257E-03  4.8878E+00  1.1600E+00  4.7246E-04  -10.35  -16.56  0079  0041
7.6377E-01  1.6969E-01 -1.8974E+01  1.2493E-01  2.0565E-02  5.2096E-03  7.2750E+00  1.6474E+00  4.9455E-04  -10.35  -14.49  0079  0044
7.6430E-01  1.2049E-01 -1.9173E+01  8.5182E-02  1.7090E-02  4.2795E-03  3.7879E+00  9.4660E-01  4.3300E-04  -12.42  -18.63  0082  0038
8.4162E-01  1.3756E-01 -1.9058E+01  9.8269E-02  1.7949E-02  5.1669E-03  4.6527E+00  1.2671E+00  4.5164E-04  -12.42  -16.56  0082  0041
8.7368E-01  1.5091E-01 -1.8891E+01  1.0841E-01  1.7925E-02  5.0000E-03  4.8505E+00  1.3107E+00  4.7948E-04  -12.42  -14.49  0082  0044
```